EDA of the authors, their publications and citations

Muhammad Taha Nasir

2022-07-27

1.Loading of libraries

The pre-installed libraries have been loaded in RStudio by calling each of them using the library function. R programming language has been used for analysis and visualization.

```
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(tidyr)

## Warning: package 'tidyr' was built under R version 4.1.3

library(readxl)
library(stringr)
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.1.3
```


2.Data import

The pre-processed data in excel has been imported into Rstudio using read_xlsx and read.csv functions as shown below:

```
papers<-read_xlsx("papers_new.xlsx")
citations<-read.csv("citations.csv")
```

3. Data Manipulation (calculation of references and citations)

The papers_new.xlsx and citations.csv been saved into papers and citations respectively. Then citations has been grouped by the cited column. Transmute function has been used to

retain the citing column and create a new column citations by using the n() which counts the no.of citations. Then ungroup function has been used. The data frame has been grouped by citing in the next step. Finally references have been calculated by using the n() function. Transmute has been used to retain the needed columns. Pipe operator has been used to join the various dplyr functions. This data frame has been named as new_citations In the last step, a new data frame ref_cit has been created by removing the duplicate values in the new_citations$cited column.
The code and ref_cit data frame has been displayed below:

```
new_citations<-citations %>%
  group_by(cited) %>%
  transmute(citations=n(),citing) %>%
  ungroup() %>%
  group_by(citing) %>%
  transmute(cited,references=n(),citations)
ref_cit<-new_citations[!duplicated(new_citations$cited),]
ref_cit

## # A tibble: 1,783 x 4
## # Groups:   citing [257]
##    citing              cited               references citations
##    <chr>               <chr>                    <int>     <int>
##  1 WOS:000752849700035 WOS:000498404700027          4         1
##  2 WOS:000752849700035 WOS:000353015800124          4         4
##  3 WOS:000752849700035 WOS:000548999400001          4         1
##  4 WOS:000752849700035 WOS:000545288600005          4         2
##  5 WOS:000758152200001 WOS:000504552000001          1         2
##  6 WOS:000745515800001 WOS:000703327100001         26         1
##  7 WOS:000745515800001 WOS:000541127800052         26         1
##  8 WOS:000745515800001 WOS:000566296200001         26         2
##  9 WOS:000745515800001 WOS:000582586400022         26         1
## 10 WOS:000745515800001 WOS:000464140500001         26         2
## # ... with 1,773 more rows
```


4. Data cleaning and joining

As far as data cleaning is concerned, na terms have been removed from Author names column of papers data frame using drop_na function in order to avoid errors in the analysis. This new data frame has been named paper_main. full_join function has been used to join the papers_main and ref_cit data frames by matching the UT and cited columns respectively. This has been saved into joined_df. is.na function has been used to check whether there are any null values in the columns of the newly joined data frames. This data frames consists of 8266 entries and 17 columns. ## Removal of na terms from Author names column

```
papers_main<-papers %>%
  drop_na(Author_Full_Names)
```

##Joining data of the the cleaned and the imported table.

```
joined_df<-papers_main %>%
  full_join(ref_cit,by=c("UT"="cited"))
```

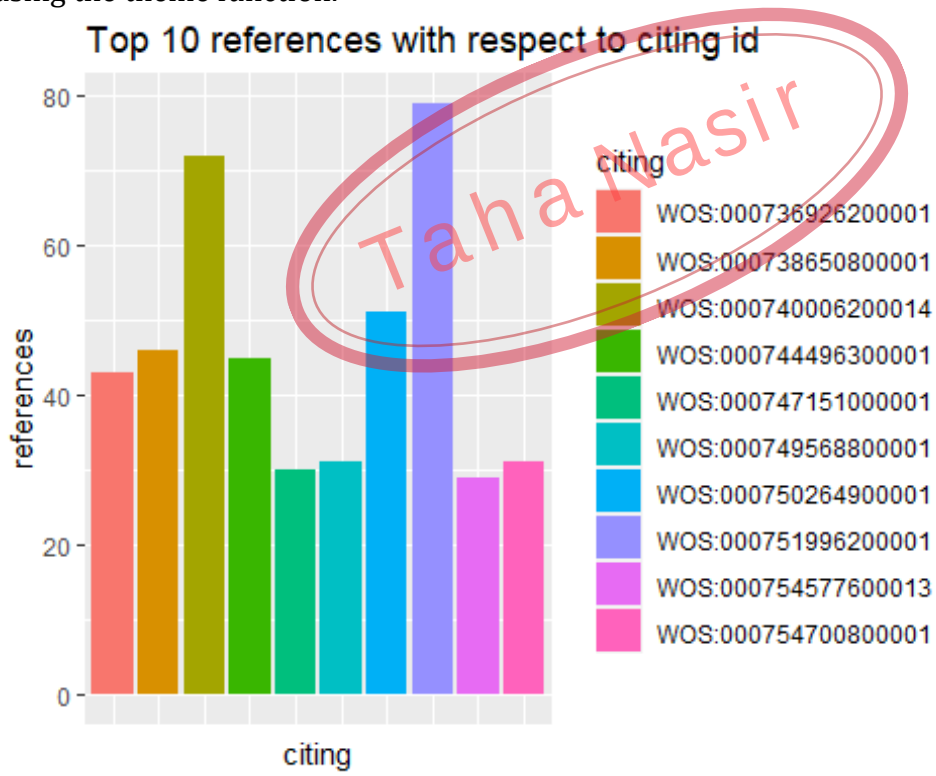
Then it is checked if there are any na in col1,2 or 3 using the following code:

```
vec=is.na(joined_df$UT)
count=sum(vec)
print(count)

## [1] 0
```

5.Citing id vs top 10 references visualization

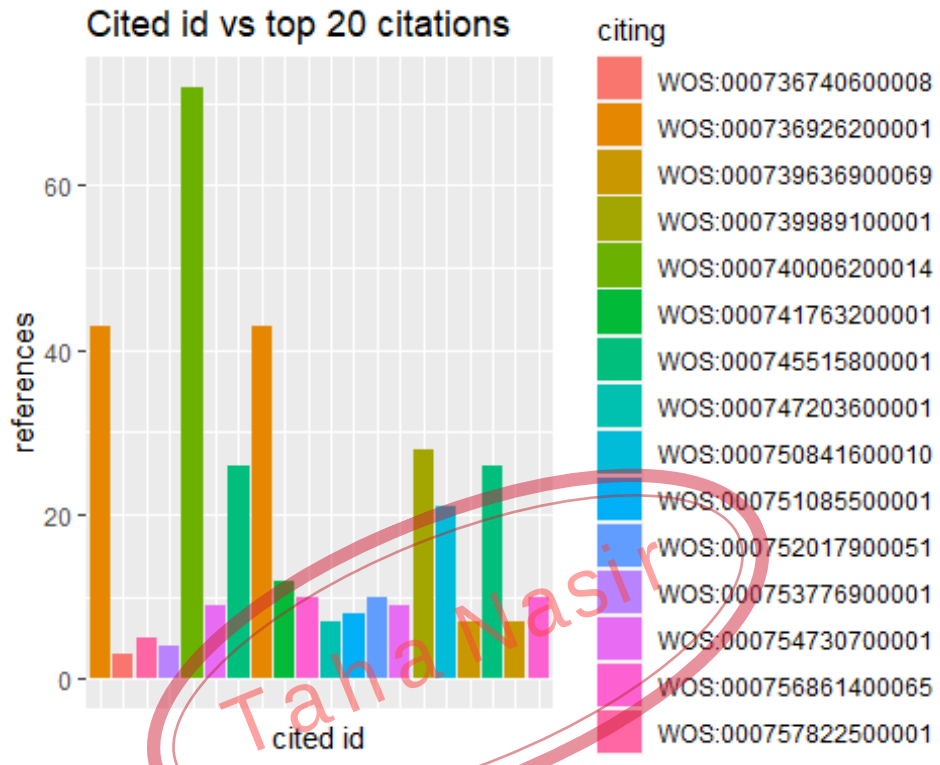
Duplicated values of ref_cit$citing has been removed and citing and references columns have been selected from the ref_cit data frame. Then references have been arranged in the descending order using the arrange function and then head(10) function has been used to keep the top 10 references. Pipe operator has been used to pipe the functions with each other. geom_bar (ggplot2 library) function has been used to display the bar graph of citing id (on the x-axis) vs references (on the y axis). The x-axis text and ticks have been removed using the theme function.



The bar graph above shows that the most no.of references were cited by the citing id WOS:0075199620001 around 79 references and the least no.of references were given by the id WOS:000754577600013 around 29 references.

6.Plotting cited id vs top 20 citations

Pipe operators have been used to combine ref_cit data frame, arrange(desc(citations)) and head(20) functions inside the ggplot function. geom_col function has been used to display the relationship between cited id and top 20 citation using a bar graph. Labs function has been used to label the title, x-axis and y-axis. x-axis ticks and text has been removed using the theme function.



The bar graph shows the the most cited id WOS:000740006200014 was cited 72 times whereas the least cited id WOS:000736740600008 was cited just 2 times among the top 20 citations. The code used to generate the plot has already been shown in the appendix portion of the ppt. So, there's no need to display it here as well. # 7. Further data cleaning and manipulation The next step is data cleaning and data joining joined_df data frame has been grouped by Author_Full_Names and mutate function is used to calculate the publications with respect to each Author. All of this has been stored in publications_df data frame. Then duplicated values of Author names have been removed from the Author_Full_Names column. Author_Full_Names and have been selected by the select function and the publications data frame has been arrange in the descending order of publication and head function has been used to display the top 10 Authors with most publications. All of this has been stored in the plt_pub_auth data frame. The code and data frame has been shown below:

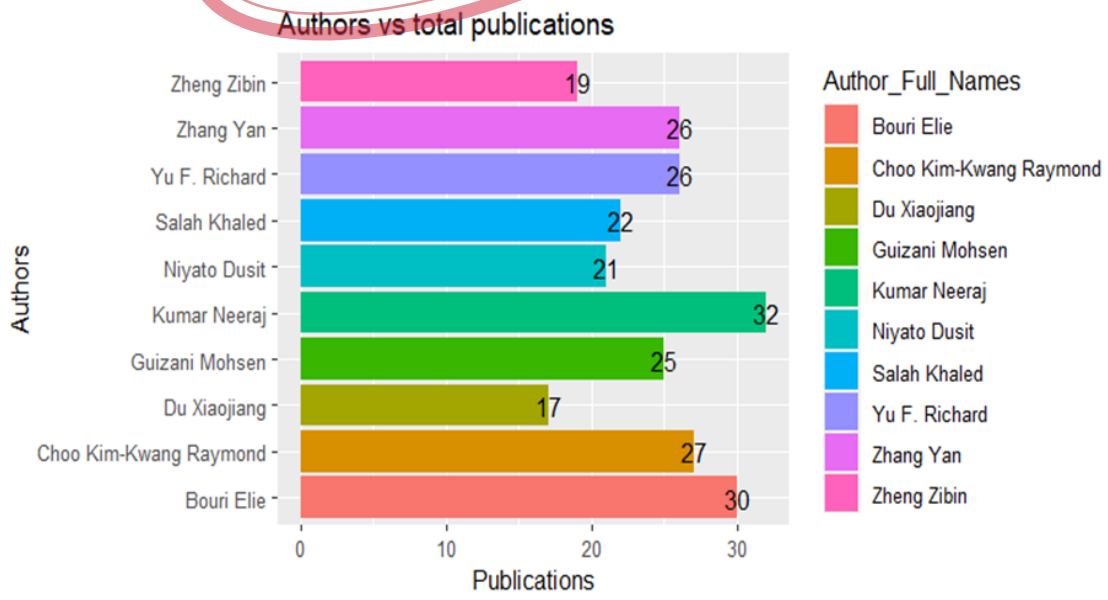
```
publications_df<-joined_df %>%
  group_by(Author_Full_Names)%>%
  mutate(publications=n())
```

```
plt_pub_auth<-publications_df[!duplicated(publications_df$Author_Full_Names),
]%>%
  select(Author_Full_Names,publications) %>%
  arrange(desc(publications))%>%
  head(10)
knitr::kable(plt_pub_auth)
```

Author_Full_Names	publications
Kumar Neeraj	32
Bouri Elie	30
Choo Kim-Kwang Raymond	27
Zhang Yan	26
Yu F. Richard	26
Guizani Mohsen	25
Salah Khaled	22
Niyato Dusit	21
Zheng Zibin	19
Du Xiaojiang	17

8.Top 10 Authors with most publications

geom_col and geom_text functions of the ggplot2 library have been used to plot the Authors vs Publications bar graph. coord_flip function has been used to flip the coordinates and labs function has been used to name the title, x-axis and y-axis. echo=FALSE argument has been used in R markdown so that only plot is displayed in the knitted file (the code has been hidden).



The plotted figure shows the top 10 author with most publications.it shows that the author with most publications (32) is Kumar Neeraj and among these authors the one with the least publications is Du Xiaojiang.

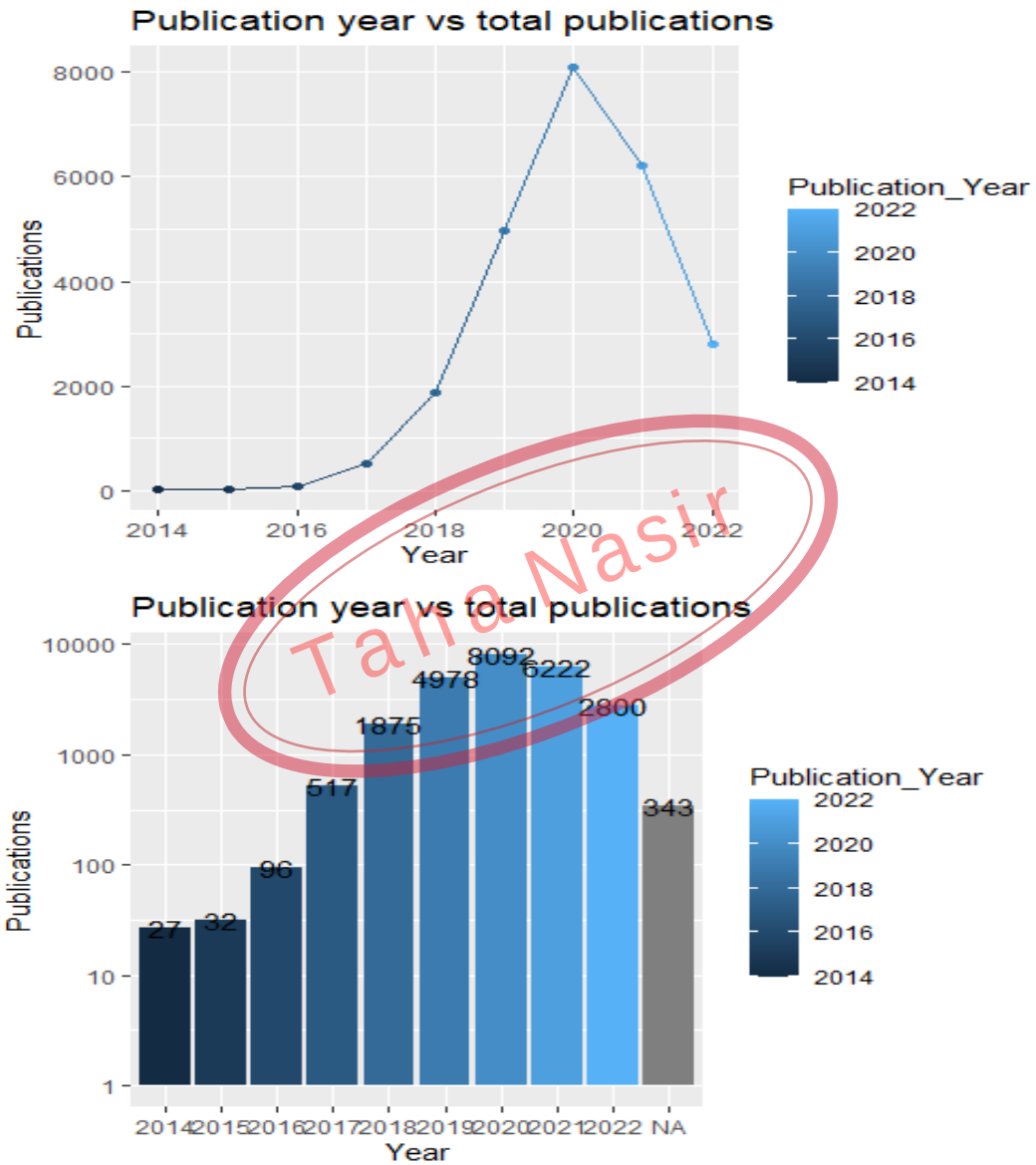
9.Publication year vs Publications plots

In the next step, ungroup function has been used on the publications_df data frame and then it has been grouped by the publication year (variable: Publication_Year). Summarize function has been used to calculate the total publications= sum(publications). The newly formed df: plotted_pub shows that 2020 was the year in which the total publication were the highest. ggplot2 library has been used to plot bar graph and line graph which shows the relationship between publication year and total publications. The graphs show that the no. of publications keep on increasing from 2014-2020 but they show a decline after 2020 till 2022.

```
plotted_pub<-publications_df %>%
  ungroup()%>%
  group_by(Publication_Year)%>%
  summarize(total_publications=sum(publications))
knitr::kable(plotted_pub)
```

Publication_Year	total_publications
2014	27
2015	32
2016	96
2017	517
2018	1875
2019	4978
2020	8092
2021	6222
2022	2800
NA	343

Then, geom_col and geom_text functions have been used to formulate the bar graph of publication year vs total publications. Names of the title, x-axis and y-axis have been given by the labs function. A logarithmic scale has been used to scale the y-axis so that we can see clear visual relationship between publication year and total publications. The data frame used is plotted_pub. Geom_point and geom_line functions have been used to plot the line graph which shows the same relationship as shown in the aforementioned bar graph. Both graphs show that 2020 was the year with most publications (exactly 8092 publications) whereas 2014 was the year with least publications (exactly 27 publications). We also have NA which means that we don't know which year this data belongs in.



Both graphs show that 2020 was the year with most publications (exactly 8092 publications) whereas 2014 was the year with least publications (exactly 27 publications). We also have NA which means that we don't know which year this data belongs in.